

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-59610

(P2000-59610A)

(43) 公開日 平成12年2月25日 (2000.2.25)

(51) Int.Cl.<sup>7</sup>

識別記号

F I

テマコード\* (参考)

H 0 4 N 1/387

H 0 4 N 1/387

5 B 0 5 7

G 0 6 T 1/00

G 0 9 C 5/00

5 C 0 7 6

G 0 9 C 5/00

G 0 6 F 15/66

B 5 J 1 0 4

9 A 0 0 1

審査請求 未請求 請求項の数10 F D (全 13 頁)

(21) 出願番号

特願平10-236315

(22) 出願日

平成10年8月7日 (1998.8.7)

(71) 出願人 000006747

株式会社リコー

東京都大田区中馬込1丁目3番6号

(72) 発明者 阿部 倭

東京都大田区中馬込1丁目3番6号 株式会社リコー内

(74) 代理人 100090240

弁理士 植本 雅治

Fターム(参考) 5B057 AA20 BA02 CC02 CG09

5C076 AA14 CA08

5J104 AA14 PA14

9A001 EE02 EE03 GG01 HH28 HZ27

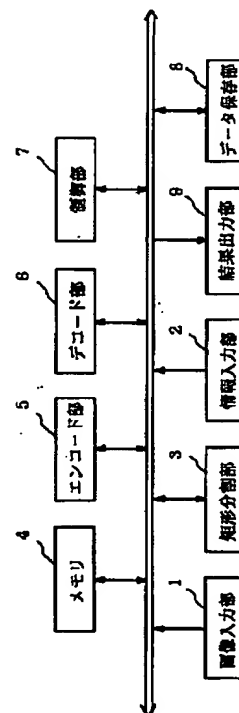
JZ63 LL01 LZ02

(54) 【発明の名称】 画像処理装置および画像処理方法および情報記録媒体

(57) 【要約】

【課題】 原画像に所定の情報を複雑な処理を必要とせずに埋め込むことができ、また、画像データに対して、その一部分が切り取られた場合でも、画像データに埋め込まれた情報を安定的に確実に抽出(デコード)できる画像処理装置および画像処理方法および情報記録媒体を提供する。

【解決手段】 デジタル画像データを所定の大きさの矩形に分割し、分割された各矩形のデジタル画像データに対し、所定の規則に基づいて所定の情報を電子透かしとして埋め込むエンコード部5と、デジタル画像データに埋め込まれている電子透かし情報をデコードするデコード部6とを有している。



【特許請求の範囲】

【請求項1】 デジタル画像データを所定の大きさの矩形に分割し、分割された各矩形のデジタル画像データに対し、所定の規則に基づいて所定の情報を電子透かしとして埋め込むエンコード手段と、デジタル画像データに埋め込まれている電子透かし情報をデコードするデコード手段とを有していることを特徴とする画像処理装置。

【請求項2】 請求項1記載の画像処理装置において、デジタル画像データに対する矩形の分割は、原画像の内容に依存せずに、原画像全体にわたってなされることを特徴とする画像処理装置。

【請求項3】 請求項1記載の画像処理装置において、デジタル画像データに対する矩形の分割は、原画像全体のうち有効な画像データを有する画素を全く含まないか非常に少数しか含まない部分を除外してなされることを特徴とする画像処理装置。

【請求項4】 請求項1記載の画像処理装置において、前記エンコード手段は、分割された矩形内のデジタル画像データに対して、矩形内を所定の大きさのウィンドウで走査し、ウィンドウ内のパターンが所定のパターンとなるとときに、ビット1またはビット0の情報を埋め込むことを特徴とする画像処理装置。

【請求項5】 請求項4記載の画像処理装置において、前記ウィンドウは、 $3 \times 3$ 画素の大きさであり、分割された矩形内のデジタル画像データに対して、該ウィンドウを3画素ずつずらして走査するようになっていることを特徴とする画像処理装置。

【請求項6】 請求項1記載の画像処理装置において、前記エンコード手段は、デコード手段による電子透かし情報のデコードの際に矩形の位置が不明なるのを防ぐために、各矩形内のデジタルデータに対して、所定の規則に基づいて、矩形位置判定用情報を電子透かしとしてさらに埋め込むことを特徴とする画像処理装置。

【請求項7】 請求項1記載の画像処理装置において、前記エンコード手段は、デコードの際の成功率を高めるために、電子透かしとして埋め込まれるべき情報に誤り訂正符号を付加することを特徴とする画像処理装置。

【請求項8】 請求項1記載の画像処理装置において、前記エンコード手段は、デコードの際の成功率を高めるために、分割された全ての矩形に、同じ電子透かし情報を埋め込んでおき、前記デコード手段は、デコードの過程において、各々の矩形からそれぞれデコードされた情報が抽出された場合に、それらを統合することを特徴とする画像処理装置。

【請求項9】 デジタル画像データを矩形に分割し、分割された各矩形内のデジタル画像データに対して、所定の大きさのウィンドウで走査し、ウィンドウ内のパターンが所定のパターンとなるとときに、所定の情報を電子透かしとして埋め込むエンコード工程と、各矩形内に埋め込まれた電子透かし情報をデコードするデコード工程と

を有していることを特徴とする画像処理方法。

【請求項10】 デジタル画像データを矩形に分割し、分割された各矩形内のデジタル画像データに対して、所定の大きさのウィンドウで走査し、ウィンドウ内のパターンが所定のパターンとなるとときに、所定の情報を電子透かしとして埋め込むエンコード工程と、各矩形内に埋め込まれた電子透かし情報をデコードするデコード工程とのうち少なくとも1つの工程を有していることを特徴とするプログラムを記録した情報記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、画像処理装置および画像処理方法および情報記録媒体に関する。

【0002】

【従来の技術】デジタル画像データは各種の媒体間でデジタルコピーすることが容易であり、特に近年、インターネットを始めとするネットワークインフラの整備や、電子図書館に代表される画像データベースの普及に伴って、デジタル画像データがデジタルコピー等によって一旦第三者に渡ってしまうと、その後の広まる範囲は非常に大きく、著作権者がデジタル画像の著作権を主張することは非常に困難である。

【0003】また、有価証券等をデジタルコピー機やスキャナで取り込むことは法律で禁止されているが、現状では予め登録可能な紙幣等しか複写を防止する手立てはない。

【0004】上記のような問題を解決するため、デジタル画像データへの電子透かし技術が着目されている。例えば特開平9-191394号には、周波数変換(具体的にはフーリエ変換)によって原画像に情報を混入する電子透かし挿入方法が示されている。

【0005】

【発明が解決しようとする課題】しかしながら、上述した周波数変換処理は一般的に処理時間が大きく実用には困難である。また一部分を切り取った場合に、画像全体の周波数特性(主に低周波)が均一であると仮定すればデコードが可能であるが、実画像では、周波数特性(主に低周波)は部分毎に異なっており、デコードは不可能であるという問題があった。

【0006】本発明は、原画像に所定の情報を複雑な処理を必要とせずに埋め込むことができ、また、画像データに対して、その一部分が切り取られた場合でも、画像データに埋め込まれた情報を安定的に確実に抽出(デコード)できる画像処理装置および画像処理方法および情報記録媒体を提供することを目的としている。

【0007】

【課題を解決するための手段】上記目的を達成するために、請求項1記載の発明は、デジタル画像データを所定の大きさの矩形に分割し、分割された各矩形のデジタル画像データに対し、所定の規則に基づいて所定の情報を

電子透かしとして埋め込むエンコード手段と、デジタル画像データに埋め込まれている電子透かし情報をデコードするデコード手段とを有していることを特徴としている。

【0008】また、請求項2記載の発明は、請求項1記載の画像処理装置において、デジタル画像データに対する矩形の分割は、原画像の内容に依存せずに、原画像全体にわたってなされることを特徴としている。

【0009】また、請求項3記載の発明は、請求項1記載の画像処理装置において、デジタル画像データに対する矩形の分割は、原画像全体のうち有効な画像データを有する画素を全く含まないか非常に少数しか含まない部分を除外してなされることを特徴としている。

【0010】また、請求項4記載の発明は、請求項1記載の画像処理装置において、前記エンコード手段は、分割された矩形内のデジタル画像データに対して、矩形内を所定の大きさのウィンドウで走査し、ウィンドウ内のパターンが所定のパターンとなるときに、ビット1またはビット0の情報を埋め込むことを特徴としている。

【0011】また、請求項5記載の発明は、請求項4記載の画像処理装置において、前記ウィンドウは、 $3 \times 3$ 画素の大きさであり、分割された矩形内のデジタル画像データに対して、該ウィンドウを3画素ずつずらして走査するようになっていることを特徴としている。

【0012】また、請求項6記載の発明は、請求項1記載の画像処理装置において、前記エンコード手段は、デコード手段による電子透かし情報のデコードの際に矩形の位置が不明になるのを防ぐために、各矩形内のデジタル画像データに対して、所定の規則に基づいて、矩形位置判定用情報を電子透かしとしてさらに埋め込むことを特徴としている。

【0013】また、請求項7記載の発明は、請求項1記載の画像処理装置において、前記エンコード手段は、デコードの際の成功率を高めるために、電子透かしとして埋め込まれるべき情報に誤り訂正符号を付加することを特徴としている。

【0014】また、請求項8記載の発明は、請求項1記載の画像処理装置において、前記エンコード手段は、デコードの際の成功率を高めるために、分割された全ての矩形に、同じ電子透かし情報を埋め込んでおき、前記デコード手段は、デコードの過程において、各々の矩形からそれぞれデコードされた情報が抽出された場合に、それらを統合することを特徴としている。

【0015】また、請求項9記載の発明は、デジタル画像データを矩形に分割し、分割された各矩形内のデジタル画像データに対して、所定の大きさのウィンドウで走査し、ウィンドウ内のパターンが所定のパターンとなるときに、所定の情報を電子透かしとして埋め込むエンコード工程と、各矩形内に埋め込まれた電子透かし情報をデコードするデコード工程とを有していることを特徴と

している。

【0016】また、請求項10記載の発明は、デジタル画像データを矩形に分割し、分割された各矩形内のデジタル画像データに対して、所定の大きさのウィンドウで走査し、ウィンドウ内のパターンが所定のパターンとなるときに、所定の情報を電子透かしとして埋め込むエンコード工程と、各矩形内に埋め込まれた電子透かし情報をデコードするデコード工程とのうち少なくとも1つの工程を有していることを特徴としている。

【0017】

【発明の実施の形態】以下、本発明の実施形態を図面に基づいて説明する。図1は本発明に係る画像処理装置の構成例を示す図である。図1を参照すると、この画像処理装置は、電子的なデジタル画像データを原画像として取り込む画像入力部1と、埋め込まれるべき所定の情報を入力する情報入力部2と、原画像を矩形に分割する矩形分割部3と、原画像(デジタル画像データ)や埋め込まれるべき所定の情報などが記憶されるメモリ4と、原画像に所定の情報を電子透かしとして埋め込むエンコード処理を行なうエンコード部5と、デジタル画像データに埋め込まれている所定の情報を抽出するデコード処理を行なうデコード部6と、各種制御を行なう制御部7と、画像データなどを保存するデータ保存部8と、デコード結果などを出力する結果出力部9とを有している。

【0018】ここで、情報入力部2は、原画像に対して電子透かしとして埋め込まれるべき所定の情報(文字列、画像など)を入力させるようになっている。

【0019】また、矩形分割部3は、原画像を予め決められた大きさ(例えば、 $255 \text{画素} \times 255 \text{画素}$ の大きさ)の矩形(ブロック)に分割するようになっている。この際、矩形分割部3は、図2に示すように、縦、横ともそれぞれ予め決定した大きさの矩形で、原画像の内容に依存せず全体を分割することもできるし、あるいは、図3に示すように、縦、横ともそれぞれ予め決定した大きさの矩形で、原画像全体のうち有効な画像データを有する画素を全く含まないか非常に少数しか含まない部分を除外して、分割することもできる。本発明では、後述のように、原画像中の文字画像などの部分を利用して情報を埋め込むようにしており、文字画像などが存在しない矩形(ブロック)では情報の埋め込みはなされない。従って、情報などを効率良く埋め込むためには、図3のように、縦、横ともそれぞれ予め決定した大きさの矩形で、原画像全体のうち有効な画像データを有する画素を全く含まないか非常に少数しか含まない部分を除外して、分割する方が良い。なお、以下では、説明の便宜上、画像入力部1から取り込まれた原画像は、2値画像であるとする。

【0020】また、エンコード部5は、情報入力部2から入力された埋め込まれるべき所定の情報をビット列に変換して、分割された各矩形(各ブロック)のそれぞれに

ついて、変換したビット列(埋め込み情報)の埋め込み処理を行なうようになっている。この埋め込み処理は、具体的には、次のようにして行なうようになっている。すなわち、埋め込み処理は、基本的には、埋め込みたい情報(ビット列とする)を $3 \times 3$ の大きさのウインドウの2次元パターンに対応させることによってなされる。

【0021】図4(a)には、ビット1を表わすパターン(4種類)が示され、また、図4(b)には、ビット0を表わすパターン(4種類)が示されている。

【0022】ここで、エンコード処理は、分割された各矩形が、例えば、 $255 \times 255$ 画素の大きさをもつとき、 $255 \times 255$ 画素の大きさの各矩形(各ブロック)のそれぞれに対して同じ情報(ビット列)の埋め込みを行なうようになっている。そして、各矩形(各ブロック)のそれぞれについて、例えば、図5に示すように、 $3 \times 3$ の大きさのウインドウを矩形の左上から右下に向けてラスタスキャン方向に3画素ずつずらしながら走査して埋め込み処理(エンコード処理)を行なうようになっている。すなわち、この例では、埋め込み周期は3画素であり、矩形(ブロック)の大きさは $255 \times 255$ 画素であって、埋め込み周期(3画素)の整数倍となっている。これにより、矩形(ブロック)の大きさは、埋め込んだ位置の周期が矩形(ブロック)の境界のところで不連続になるのを防止できる。

【0023】図6(a)乃至(d)、図7(a)乃至(d)、図8(a)乃至(d)は、埋め込み処理のルール(規則)を示す図である。埋め込み処理のルール(規則)は、次のようになっている。すなわち、原画像(2値画像)を $3 \times 3$ のウインドウで走査するときに、例えば、図6(a)乃至(d)のパターン $P_0$ のいずれかが出現するとき、埋め込むべき情報がビット1のときには、パターン $P_0$ をパターン $P_1$ のように変換し、また、埋め込むべき情報がビット0のときには、パターン $P_0$ をパターン $P_2$ のように変更する。

【0024】また、原画像(2値画像)を $3 \times 3$ のウインドウで走査するときに、例えば、図7(a)乃至(d)のパターン $P_0$ のいずれかが出現するとき、埋め込むべき情報がビット1のときには、パターン $P_0$ はパターン $P_1$ となっているので、パターン $P_0$ を変更せず、そのままにし、また、埋め込むべき情報がビット0のときには、パターン $P_0$ をパターン $P_2$ のように変更する。

【0025】また、原画像(2値画像)を $3 \times 3$ のウインドウで走査するときに、例えば、図8(a)乃至(d)のパターン $P_0$ のいずれかが出現するとき、埋め込むべき情報がビット1のときには、パターン $P_0$ をパターン $P_1$ のように変更し、また、埋め込むべき情報がビット0のときには、パターン $P_0$ はパターン $P_2$ となっているので、パターン $P_0$ を変更せず、そのままにする。

【0026】このように、 $3 \times 3$ のウインドウを図5に示したような仕方で走査していき、埋め込む対象となる

特定のパターン $P_0$ が出現したときには、埋め込むべきビット値(ビット1またはビット0)に応じてこのパターン $P_0$ を変更することで、ビット1、ビット0の埋め込みを行なうことができる。この際、上記のルール(規則)では、原画像(2値画像)の主にエッジ部分に埋め込むことができ、2次元的なトポロジーが変換せず、見た目に気付かれにくくなっている。

【0027】図9には、埋め込み情報(電子透かし)としてのビット列、例えば”0010111…”を各矩形(各ブロック)のそれぞれについて埋め込んだ状態の一例が示されている。ここで、各矩形(各ブロック)には、同じビット列”0010111…”が埋め込まれるが、その埋め込まれ方は、各矩形(各ブロック)内の2値画像データの状態によって変わる。例えば、1つの矩形(ブロック)内に文字画像などが少ない場合には、ビット列”0010111…”は、この矩形(ブロック)内でまばらに埋め込まれる。これに対し、1つの矩形(ブロック)内に文字画像などが多く存在する場合には、ビット列”0010111…”は、この矩形(ブロック)内で密に埋め込まれる。なお、図9において、“1”、“0”が記されている箇所は、図6、図7、図8のパターン $P_0$ が存在したところを意味し、“1”、“0”が記されていない箇所は、パターン $P_0$ 以外のパターンとなっているところ(“1”、“0”が埋め込まれない箇所)を意味している。

【0028】ところで、後述のデコード処理時に、各矩形ごとに埋め込まれた情報(ビット列)をデジタル画像データから取り出すためには、上述のように分割された各矩形(各ブロック)の先頭位置が正しく割り出される必要がある(デコードの際にオリジナルの矩形の位置が不明になるのを防ぐ必要がある)。すなわち、分割された少なくとも1つの矩形を包括するように画像の一部分が切り取られればデコードが可能となるが、一部分を切り取った際に、特に問題となるのは、画像をどの部分で矩形に分割したかが不明になることである。なお、画像データの場合、全体の中から欲しい一部分のみを切り取って利用することは日常的に行なわれることであるが、従来の電子透かし方法ではこのような画像処理に対しては無力であった。

【0029】この課題を解決するため、エンコード部5は、各矩形(各ブロック)の先頭位置を識別させるための情報(矩形位置判定用情報)、すなわちビット列として例えば”11111111”を、埋め込まれるべき本来の情報(例えば”0010111…”)に付加するようになっている(埋め込まれるべき本来の情報(例えば”0010111…”)に先立って埋め込むようになっている)。

【0030】図10には、矩形位置判定用情報として、ビット列”11111111”が各矩形(各ブロック)に付加された状態の一例が示されている。なお、この場合も、前述したように、各矩形(各ブロック)には、同じビ

ット列”111111110010111…”が埋め込まれるが、その埋め込まれ方は、各矩形(各ブロック)内の2値画像の状態によって変わる。

【0031】さらに、エンコード部5は、デコードの際の成功率を高めるために、埋め込まれるべき所定の情報に誤り訂正符号を付加することができる。

【0032】また、デコード部6は、エンコード部5で上記のように原画像に埋め込まれた情報(電子透かし情報)を読み出すようになっている。すなわち、エンコード部5のエンコード処理に合わせて、3×3の大きさのウィンドウをラスタスキャン方向に3画素ずつずらしながら走査して、図4(a)のいずれかのパターンが出現したとき、ビット1と判断し、図4(b)のいずれかのパターンが出現したときに、ビット0と判断して、埋め込まれたビット列をデコードするようになっている。

【0033】この際、第三者が、例えば図10に示したような画像に対し、図11に示すように、符号Dの部分を取り取って無断で用いたとする。この場合、著作権者は、第三者によって用いられている符号Dの部分に対してデコード部6でデコード処理して、この画像に所定の情報(電子透かし)が埋め込まれているか否かを調べるが、上記符号Dの部分は、一般には、矩形の区切りとは一致しておらず、このままでは、矩形(ブロック)内に埋め込まれたビット列を正しく読み出すことができない。

【0034】そこで、符号Dの部分に対してデコード処理を行なうとき、デコード部6は、矩形(ブロック)の先頭位置を先ず検出するようになっている。すなわち、エンコード部5では、前述のように、矩形(ブロック)の先頭位置を識別させるために、図10に示したように、矩形位置判定用情報として、ビット列”11111111”を各矩形(各ブロック)に付加しているので、デコード部6は、デコード対象となる画像をラスタスキャン方向に3×3の大きさのウィンドウで順次に走査してデコードするとき、ビット列”11111111”が得られるか否かを判断し、ある走査位置となったときにビット列”11111111”が得られたときには、この位置を矩形(ブロック)の先頭位置として割り出すようになっている。

【0035】図12には、この様子が示されている。すなわち、符号Dの部分に対してデコード処理を行なうとき、ウィンドウが符号E<sub>1</sub>の位置となったときに、この位置E<sub>1</sub>を1つの矩形の先端位置として割り出し、さらに、ウィンドウが符号E<sub>2</sub>の位置となったときに、この位置E<sub>2</sub>を次の矩形の先端位置として割り出すようになっている。

【0036】ところで、図11のように符号Dの部分を切り取るとき、切り取られた符号Dの部分から3×3の大きさのウィンドウで3画素ずつずらしたから順次に走査する場合、この3×3の大きさのウィンドウが矩形(ブロック)の先頭の3×3の画素のところに合致すると

は限らない。図13には、符号Dの部分の切り取られ方の例(9種類の例)が示されている。図13(i)のように切り取られるときには、切り取られた符号Dの部分から3×3の大きさのウィンドウで3画素ずつずらしたから順次に走査するとき、この3×3の大きさのウィンドウが矩形(ブロック)の先頭の3×3の画素のところに合致し、従って、このときには正しいデコードを行なうことができる。しかしながら、図13(a)乃至(h)のように切り取られるときには、切り取られた符号Dの部分から3×3の大きさのウィンドウで3画素ずつずらしたから順次に走査するとき、この3×3の大きさのウィンドウが矩形(ブロック)の先頭の3×3の画素のところに合致せず、このときには正しいデコードを行なうことができない。

【0037】図13(a)乃至(i)のうちのどの仕方で切り取られるかは一般にわからないため、デコード部6は、図11のように符号Dの部分が切り取られるとき、この符号Dの部分について3×3の大きさのウィンドウの走査開始位置を図14(a)乃至(i)の9通りに設定し、9通りのそれぞれの場合についてデコード処理を試みる。そして、9通りのデコード処理において、切り取られた符号Dの部分から3×3の大きさのウィンドウで3画素ずつずらしたから順次に走査するとき、この3×3の大きさのウィンドウが矩形(ブロック)の先頭の3×3の画素のところに合致するときには、“11111111…”が得られる筈であり、このときのデコード処理結果を採用することができる。例えば、図13(a)に示すような切り取られ方がされた場合、図14(a)乃至(i)の9通りについてそれぞれデコード処理が試される時、図14(i)の場合に“11111111…”が得られ、これがデコード処理結果として採用される。

【0038】また、デコード部6において、矩形(ブロック)の先頭位置を割り出す場合、矩形(ブロック)内に対するビット列”11111111”の出現位置は、この矩形(ブロック)内の文字画像などの状態に応じて、一般に、矩形(ブロック)毎に異なったものとなり、従って、ある矩形(ブロック)ではその先頭位置を正しく割り出すことができるが、ある矩形(ブロック)ではその先頭位置を正しく割り出すことができない場合が生じる。そこで、デコード部6は、デコード対象となる画像部分Dが複数の矩形(ブロック)を含む大きさのものである場合、デコード処理を1つの矩形(ブロック)だけでなく、デコード対象となる画像部分Dに含まれている複数の矩形(ブロック)のそれぞれについて行ない、複数の矩形(ブロック)のそれぞれについてなされたデコード処理結果を統合して、最終的なデコード処理結果とするようになっている。

【0039】この統合処理としては、例えば、多数決処理を用いることができる。具体的に、例えば、ビット毎に多数決論理を用い、より信頼性の高い方のビット値を

採用するようになっている。

【0040】次にこのような構成の画像処理装置の処理動作について説明する。図15はエンコード部5におけるエンコード処理の概要を示すフローチャートである。図15を参照すると、まず、ステップS101では、情報入力部2により、埋め込まれるべき所定の情報(文字列、画像など)をメモリ4に取り込む。次に、ステップS102では、メモリ4に取り込まれた所定の情報をビット列に変換する。次に、ステップS103では、ビット列に誤り訂正符号(誤り訂正用のビット)を付加する。次に、ステップS104では、ビット列にさらに矩形位置判定用のビットを付加する。

【0041】次に、ステップS105では、所定の情報を埋め込む対象の画像(原画像)を画像入力部1によりデジタル画像データとしてメモリ4に取り込む。次に、ステップS106では、ステップS105で取り込んだ原画像を矩形に分割する。より具体的には、図2に示すように、原画像全体を画像の内容に依らずに均一に分割するか、あるいは、図3に示すように、原画像全体のうち有効な画像データを有する画素を全く含まない部分、非常に少数しか含まない部分を除外して、分割する。ここで、分割された矩形数を $N_r$ とする。

【0042】次に、ステップS107では、矩形の数を計数するカウンタ $i$ を0に初期設定する。次に、ステップS108では、カウンタ $i$ を1だけインクリメントする。次に、ステップS109では、注目矩形内を $3 \times 3$ 画素のウィンドウで3画素ずつ走査し、注目矩形内の画像データに所定の情報を埋め込む。次に、ステップS110では、カウンタ $i$ が $N_r$ よりも大きくなったかを判定し、カウンタ $i$ が $N_r$ よりも大きくなければステップS108に戻り、次の矩形について同様の埋め込み処理を行なう。そして、ステップS110で、カウンタ $i$ が $N_r$ となったとき、エンコード処理を終了する。

【0043】また、図16、図17はデコード部6におけるデコード処理の概要を示すフローチャートである。図16、図17を参照すると、まず、ステップS201では、画像入力部1により、デコードする対象の画像をデジタル画像データとしてメモリ4に取り込む。次に、ステップS202では、ステップS201で取り込んだ画像全体から埋め込まれている所定の情報を抽出し、2次元のビット列を取り出す。

【0044】次に、ステップS203では、ステップS202で得られた2次元のビット列の始点に、エンコード時に用いたウィンドウ( $3 \times 3$ 画素の大きさのウィンドウ)と同じ大きさのウィンドウをセットする。次に、ステップS204では、エンコードした時と同じ順序でウィンドウを走査し、ウィンドウ内の2次元ビット列を取り出し、1次元のビット列に変換する。

【0045】なお、2次元ビット列を介さずに直接1次

元のビット列を取り出すことが可能である。ただし、処理時間を考慮した場合、画像を一旦2次元のビット列に置き換えてから1次元ビット列を取り出したほうが有利になる。つまり、2次元のビット列を介さない場合、毎回画像を走査しながらビット値を取り出すことになる。それに対して、一旦2次元のビット列に置き換えると、そのサイズは画像サイズの $1/3 \times 1/3 = 1/9$ になり、高速化が望める。

【0046】次に、ステップS205では、ステップS204で取り出した1次元のビット列がエンコード時に付加した矩形位置判定用のビットの条件を満たすか否かを判定する。この条件を満たす場合は、ステップS206へ進み、満たさない場合はステップS208へ進む。

【0047】ステップS206では、ステップS205においてステップS204で取り出した1次元のビット列がエンコード時に付加した矩形位置判定用のビットの条件を満たすと判定されたので、ステップS204で取り出した1次元のビット列を1つの候補(1つの矩形)として保存する。次に、ステップS207では、ビット列候補数(矩形数)をインクリメントして、ステップS208に進む。

【0048】ステップS208では、画像の終点までウィンドウで走査したか否かを判定する。終点まで走査していない場合はステップS209へ進み、ステップS209では、2次元ビット列上のウィンドウをラスタ方向に移動してステップS204に戻り、同様の処理を繰り返す。このようにして、ステップS208において、終点まで走査した場合はステップS210へ進む。

【0049】ステップS210では、1次元ビット列候補数(矩形数)を判定する。この結果、候補数(矩形数)が1の場合はステップS212へ進み、また、候補数(矩形数)が2以上の場合はステップS211へ進む。ステップS211では、複数のビット列候補を1つにまとめる(統合する処理を行なう)。より具体的には、例えば、ビット毎に多数決論理を用い、より信頼性の高い方のビット値を採用する。図18にはその一例が示されている。

【0050】ステップS212では、ビット列の誤り訂正符号を用いて誤り訂正を行なう。次に、ステップS213では、デコード結果を出力し、デコード処理を終了する。

【0051】このように、本発明によれば、予め画像データに著作権者の署名など任意のデータを元の画像データを損なうことなく埋め込んでおくことができ、コピーされた画像データに対してデコード処理を施すことで埋め込んだ署名などを取り出し、著作権を主張することができる。

【0052】より具体的に、本発明によれば、画像データに任意のデータ(例えば複写機の機番や日付など)をユ

ーザに気付かれることなく埋め込んでおくことができるので、デジタルデータである限りは埋め込んだデータをもとに追跡が可能である。さらに画像データに秘密の情報を電子透かしとして埋め込んで通信するといった秘匿通信などの様々な利用法が考えられ、非常に重要かつ応用の広い技術に適用できる。

【0053】上述の例では、画像入力部1から入力される原画像は、2値画像であるとしたが、本発明は画像のデータの種類によらず適用が可能である。例えば、原画像が多値、カラーなどである場合にも適用可能である。図19には、原画像が多値画像である場合の本発明の適用例が示されている。原画像が多値画像である場合、多値画像は、階調の数に応じたビット数分のビットプレーンとして表現される。図19には、多値画像が256階調のものであり(1画素当り8ビットで表現されるものであり)、従って、8枚のビットプレーンP<sub>1</sub>~P<sub>8</sub>によって表現される場合が示されている。この場合、各ビットプレーンP<sub>1</sub>~P<sub>8</sub>は、それぞれ、2値画像であるので、8枚のビットプレーンのうちの1枚のビットプレーン(2値画像)、例えばP<sub>1</sub>にエンコード部5により前述したような仕方で所定の情報を埋め込むことができる。そして、この場合には、デコード部6は、8枚のビットプレーンのうちの1枚のビットプレーン(2値画像)、例えばP<sub>1</sub>に埋め込まれている所定の情報を、前述のような仕方で読み出すことができる。

【0054】また、上述の例では、ウインドウの大きさを3×3画素としたが、ウインドウの大きさは、任意のもの、例えば2×2画素、あるいは4×4画素のものにすることができる。しかしながら、図4、図6、図7、図8で説明したように、ビット“1”、“0”を表現するためのパターンとしては、2×2画素では情報量が少なすぎ、また、4×4画素では情報量が多すぎるので、3×3画素のものが最適なパターンと考えられる。従って、ウインドウの大きさは、3×3画素であるのが最も良い。

【0055】図20は図1の画像処理装置のハードウェア構成例を示す図である。図20を参照すると、この画像処理装置は、例えばパーソナルコンピュータ等で実現され、全体を制御するCPU21と、CPU21の制御プログラム等が記憶されているROM22と、CPU21のワークエリア等として使用されるRAM23と、原画像を読み込むスキャナ24と、所定の情報を入力する入力装置25と、デコード結果などを出力する結果出力装置(例えば、ディスプレイやプリンタ)26とを有している。

【0056】ここで、スキャナ24、入力装置25、結果出力装置26は、図1の画像入力部1、情報入力部2、結果出力部6にそれぞれ対応している。また、CPU21は、図1の制御部7、矩形分割部3、エンコード部5、デコード部6の機能を有している。

【0057】なお、CPU21におけるこのような制御部7、矩形分割部3、エンコード部5、デコード部6等としての機能は、例えばソフトウェアパッケージ(具体的には、CD-ROM等の情報記録媒体)の形で提供することができ、このため、図16の例では、情報記録媒体30がセットさせるとき、これを駆動する媒体駆動装置31が設けられている。

【0058】換言すれば、本発明の画像処理装置は、イメージスキャナ、ディスプレイ等を備えた汎用の計算機システムにCD-ROM等の情報記録媒体に記録されたプログラムを読み込ませて、この汎用計算機システムのマイクロプロセッサにエンコード処理、デコード処理を実行させる装置構成においても実施することが可能である。この場合、本発明のエンコード処理、デコード処理を実行するためのプログラム(すなわち、ハードウェアシステムで用いられるプログラム)は、媒体に記録された状態で提供される。プログラムなどが記録される情報記録媒体としては、CD-ROMに限られるものではなく、ROM、RAM、フレキシブルディスク、メモリカード等が用いられても良い。媒体に記録されたプログラムは、ハードウェアシステムに組み込まれている記憶装置、例えばハードディスク装置にインストールされることにより、このプログラムを実行して、エンコード処理機能、デコード処理機能を実現することができる。

【0059】

【発明の効果】以上に説明したように、請求項1乃至請求項10の発明によれば、デジタル画像データを所定の大きさの矩形に分割し、分割された各矩形のデジタル画像データに対し、所定の規則に基づいて所定の情報を電子透かしとして埋め込むエンコード手段と、デジタル画像データに埋め込まれている電子透かし情報をデコードするデコード手段とを有しているので、原画像に所定の情報を複雑な処理を必要とせずに埋め込むことができる。

【0060】特に、請求項2の発明によれば、画像をその内容によらず全体を分割するので、処理が簡単で高速に処理することができる。

【0061】また、請求項3の発明によれば、スキャナで画像を取り込んだ場合に周囲に現れることが多い余白部分や、その他の画像中に現れる無用の余白部分を除いて矩形に分割することができる。すなわち、余白部分のみを切り取って再利用することは通常考えづらく、また情報を埋め込むにも非可視性の点から困難である。従って、余白部分を除外して情報を埋め込むことは無用な部分に情報を埋め込んで目立たせてしまうことを防止でき、無駄なエンコード処理を行わずに済む。

【0062】また、請求項6の発明によれば、請求項1記載の画像処理装置において、前記エンコード手段は、デコード手段による電子透かし情報のデコードの際に矩形の位置が不明になるのを防ぐために、各矩形内のデジ



タルデータに対して、所定の規則に基づいて矩形位置判定用情報を電子透かしとしてさらに埋め込むので、デコードの際に2次的に得られるビット列パターンから本来の埋め込んだ1次元のビット列パターンを得ることができる。すなわち、画像の一部分を切り取って再利用された場合にも埋め込んだ情報の復号(デコード)が可能となる。すなわち、画像データに対して、その一部分が切り取られた場合でも、画像データに埋め込まれた情報を安定的に確実に抽出(デコード)できる。

【0063】また、請求項7の発明によれば、請求項1記載の画像処理装置において、前記エンコード手段は、デコードの際の成功率を高めるために、電子透かしとして埋め込まれるべき情報に誤り訂正符号を付加するので、仮にいくらかの埋め込み情報の欠落があっても確実にデコードすることができる。

【0064】また、請求項8の発明によれば、デコードの対象の画像が複数の矩形をまたがって切り取られた場合に、それぞれ矩形に対してデコードされた結果が得られるが、これらの複数のデコード結果を統合することによって、より正確に埋め込み情報のデコードができる。

#### 【図面の簡単な説明】

【図1】本発明に係る画像処理装置の構成例を示す図である。

【図2】矩形分割部による矩形の分割の一例を示す図である。

【図3】矩形分割部による矩形の分割の一例を示す図である。

【図4】ビット1、ビット0を表わすパターンを示す図である。

【図5】埋め込み処理(エンコード処理)の仕方を説明するための図である。

【図6】埋め込み処理のルールを示す図である。

【図7】埋め込み処理のルールを示す図である。

【図8】埋め込み処理のルールを示す図である。

【図9】埋め込み情報(電子透かし)としてのビット列を各矩形(各ブロック)のそれぞれについて埋め込んだ状態の一例を示す図である。

【図10】矩形位置判定用情報としてのビット列”11111111”が各矩形(各ブロック)に付加された状態

の一例を示す図である。

【図11】図10の画像に対して切り取られる部分Dを示す図である。

【図12】矩形の先端位置の検出の仕方を説明するための図である。

【図13】画像の切り取られ方の例を示す図である。

【図14】9通りのデコード処理を説明するための図である。

【図15】エンコード部におけるエンコード処理の概要を示すフローチャートである。

【図16】デコード部におけるデコード処理の概要を示すフローチャートである。

【図17】デコード部におけるデコード処理の概要を示すフローチャートである。

【図18】デコード結果の統合処理の一例を示す図である。

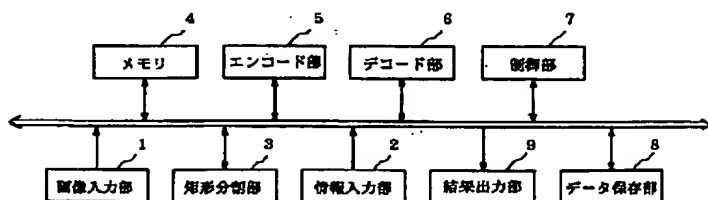
【図19】原画像が多値画像である場合の本発明の適用例を説明するための図である。

【図20】図1の画像処理装置のハードウェア構成例を示す図である。

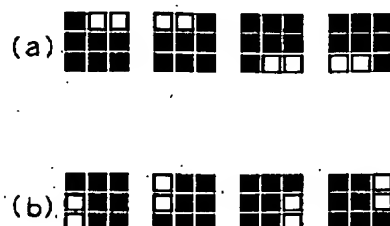
#### 【符号の説明】

|    |        |
|----|--------|
| 1  | 画像入力部  |
| 2  | 情報入力部  |
| 3  | 矩形分割部  |
| 4  | メモリ    |
| 5  | エンコード部 |
| 6  | デコード部  |
| 7  | 制御部    |
| 8  | データ保存部 |
| 9  | 結果出力部  |
| 21 | CPU    |
| 22 | ROM    |
| 23 | RAM    |
| 24 | スキャナ   |
| 25 | 入力装置   |
| 26 | 結果出力装置 |
| 30 | 情報記録媒体 |
| 31 | 媒体駆動装置 |

【図1】

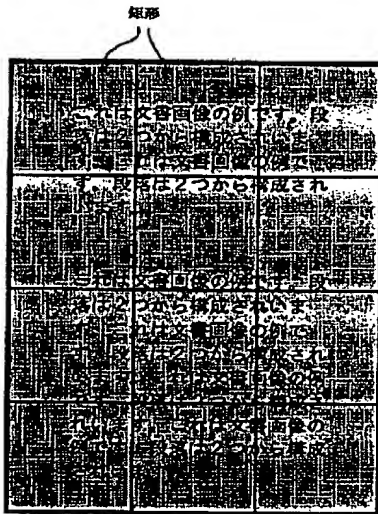


【図4】

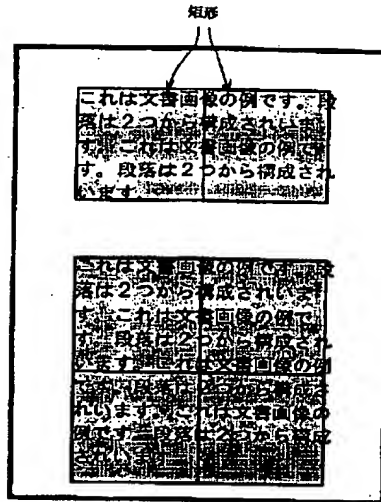




【図2】



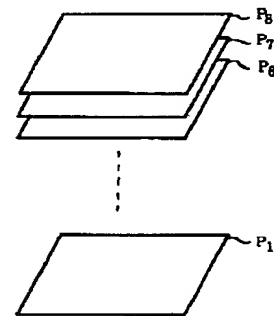
【図3】



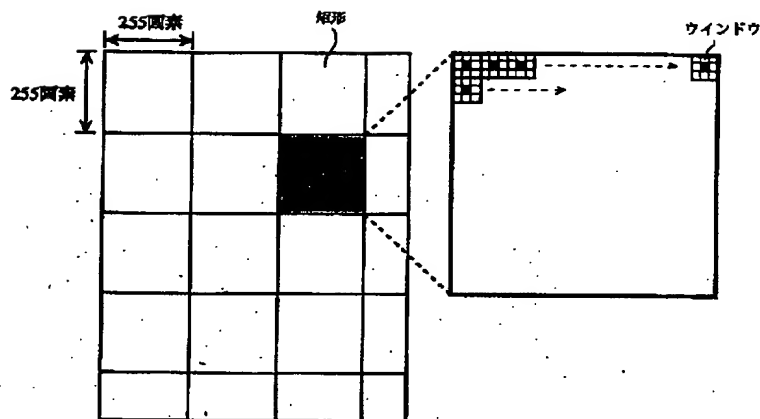
【図18】

|          |                    |
|----------|--------------------|
| 正解       | 101010101010101010 |
| デコード結果1  | 101110101010100010 |
| デコード結果2  | 101010101010101010 |
| デコード結果3  | 101010101110101010 |
| ビット毎の多数決 | 101010101010101010 |

【図19】

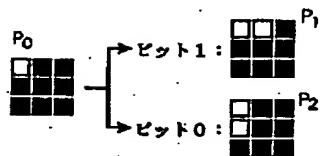


【図5】

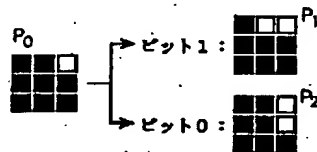


【図6】

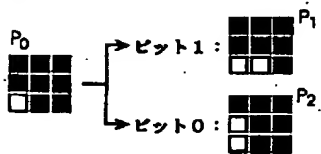
(a)



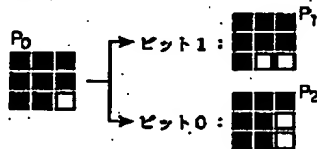
(b)



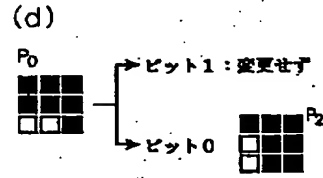
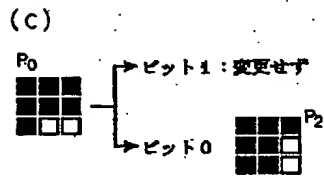
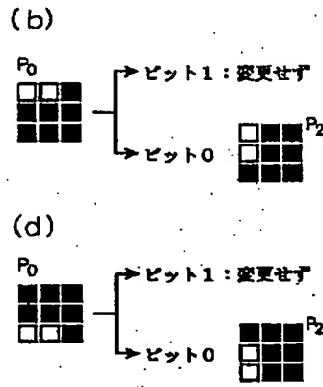
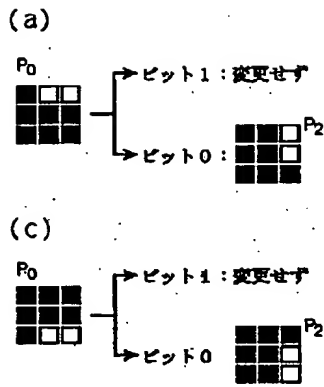
(c)



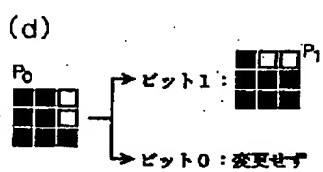
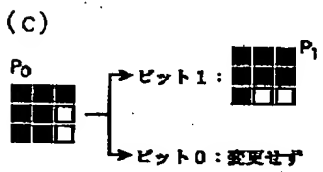
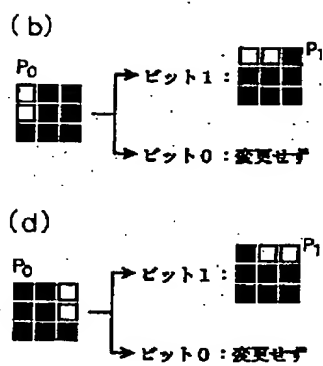
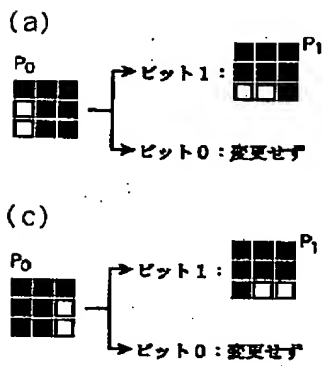
(d)



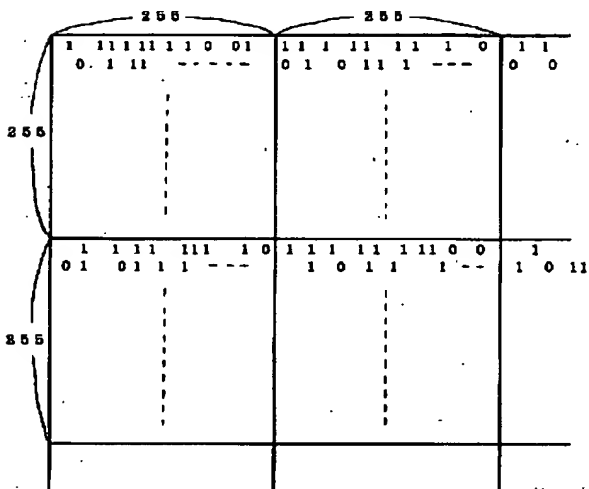
【図7】



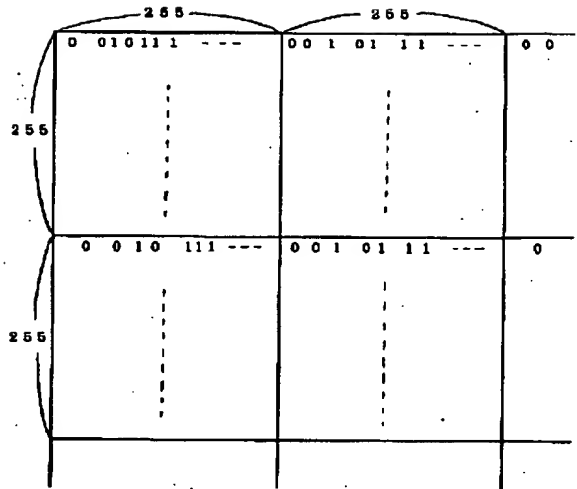
【図8】



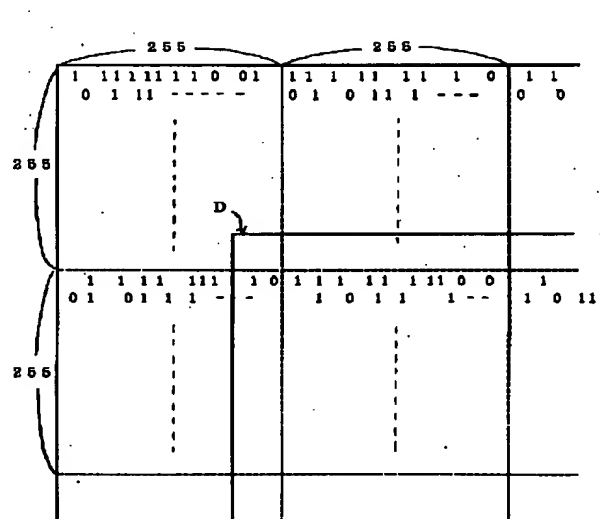
【図10】



【図9】



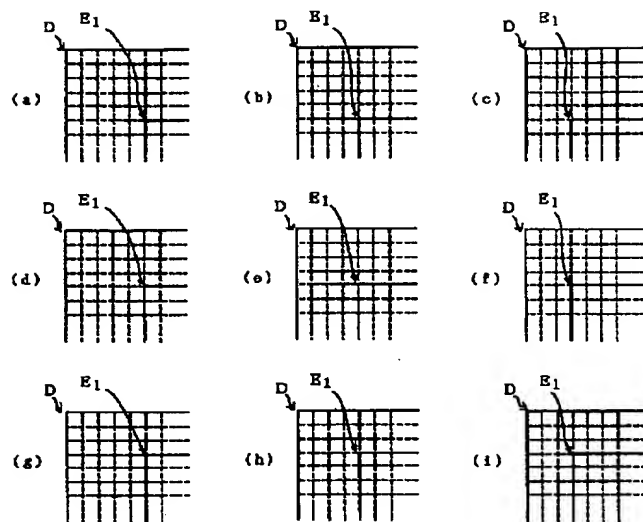
【図11】



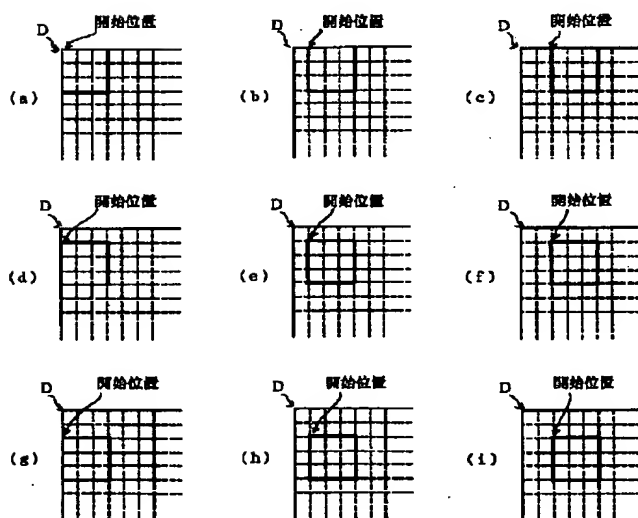
【图 1 2】

|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D | E <sub>1</sub> |   |   |   |   |   |   |   |   |   |   |   |   |   |   | E <sub>2</sub> |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   | 1              | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1              | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
|   | 0              | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0              | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|   | 1              | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1              | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
|   | 0              | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1              | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

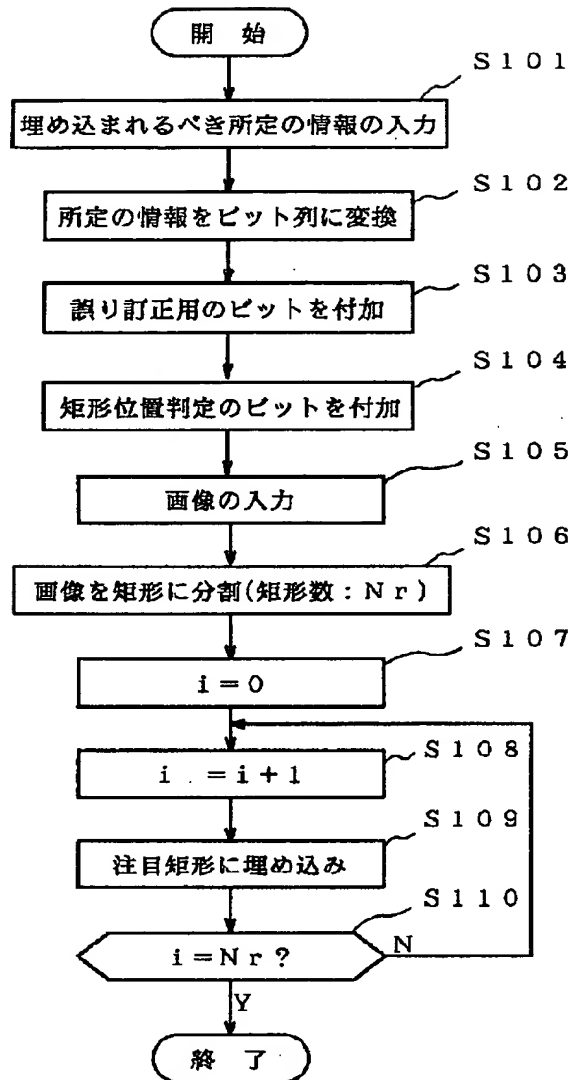
【图 1 3】



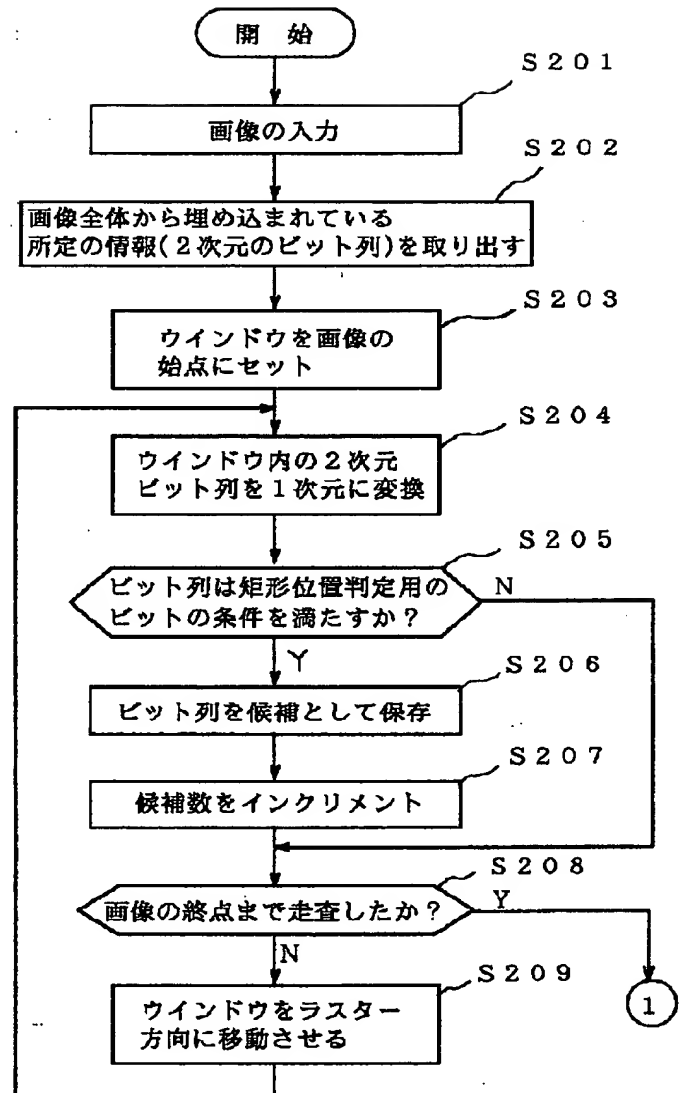
【图 1 4】



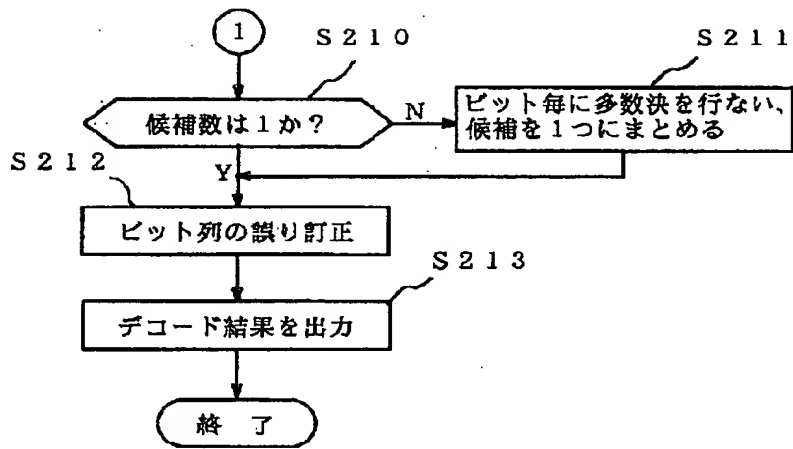
【図15】



【図16】



【図17】



【図20】

